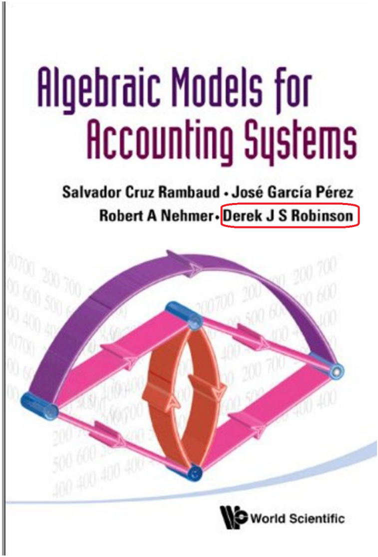
# Algebraic Models for Accounting Systems

Fatemeh Koorepazan-Moftakhar

Comenius University, Bratislava, Slovakia

Seminar on Algebraic Graph Theory, Bratislava, Slovakia


Algebraic Models for
Accounting Systems
Salvador Cruz Rambaud • José García Pérez
Robert A Nehmer • Derek J S Robinson
World Scientific

Order Number 8823212

**Accounting information systems as algebras and first order axiomatic models**

Nehmer, Robert Alan, Ph.D.

University of Illinois at Urbana-Champaign, 1988



U·M·I
300 N. Zeeb Rd.
Ann Arbor, MI 48106

Annals of Operations Research 71(1997)179–198 179

# An algebraic model for the representation of accounting systems

Robert A. Nehmer

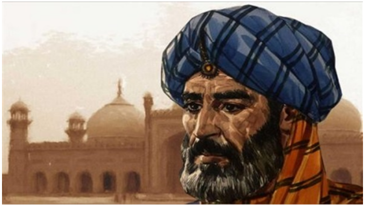
*Campbell School of Business, Berry College, Mount Berry, GA 30149–5624, USA*

Derek Robinson

*Department of Mathematics, University of Illinois-Urbana, Urbana, IL 61801, USA*

The paper describes an algebraic structure which embodies the essential features of the double-entry accounting system. The structure has the benefits of providing reliable means to record the balances of the accounts of the system and to apply transactions to the accounts. It will detect transactions which are of an undesirable type or which lead to inadmissible balances, thus preserving the integrity of the system. The stucture is also able to generate reports and includes procedures to verify whether an existing balance has been obtained by legitimate transactions. Finally, it provides methods for comparing accounting systems with one another and over time.

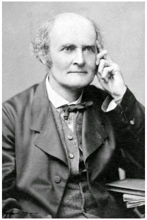
# Ibn Taymiyyah



*Accounting systems used by Muslims as early as the seventh century.*

# Franciscan monk and mathematician Luca Pacioli (1445-1517)

"Summa de Arithmetica, Geometria, Proportioni et Proportionalita"



*The principles of book-keeping by double entry constitute a theory which is mathematically by no means uninteresting; it is in fact, like Euclid's theory of ratios, an absolutely perfect one, and it is only its extreme simplicity which prevents it from being as interesting as it would otherwise be.*

- Cayley, A. (1894). The Principle of Bookkeeping by Double Entry. Cambridge University Press.

# Ellerman

*Double-entry book-keeping illustrates one of the most astonishing examples of intellectual insulation between disciplines, in this case, between accounting and mathematics.*

There are four principal structures which are used repeatedly and which appear well suited to application in accounting, namely:

- balance vector;

There are four principal structures which are used repeatedly and which appear well suited to application in accounting, namely:

- balance vector;
- directed graph (or digraph);

There are four principal structures which are used repeatedly and which appear well suited to application in accounting, namely:

- balance vector;
- directed graph (or digraph);
- automaton;

There are four principal structures which are used repeatedly and which appear well suited to application in accounting, namely:

- balance vector;
- directed graph (or digraph);
- automaton;
- monoid.

# Balance Vectors

They begin the task of assembling the various components of their basic algebraic model of an accounting system.

They begin the task of assembling the various components of their basic algebraic model of an accounting system.

The first concern is to provide *a means of describing the state of an accounting system* at any instant. Now this is most naturally accomplished by listing the “values” of the various accounts in the system.

They begin the task of assembling the various components of their basic algebraic model of an accounting system.

The first concern is to provide *a means of describing the state of an accounting system* at any instant. Now this is most naturally accomplished by listing the “values” of the various accounts in the system.

So the first step must be to identify an algebraic structure to which the account values will belong.

They begin the task of assembling the various components of their basic algebraic model of an accounting system.

The first concern is to provide *a means of describing the state of an accounting system* at any instant. Now this is most naturally accomplished by listing the “values” of the various accounts in the system.

So the first step must be to identify an algebraic structure to which the account values will belong.The structure chosen must be sufficiently rich to accommodate the operations that one would expect to apply to the accounting system.

They begin the task of assembling the various components of their basic algebraic model of an accounting system.

The first concern is to provide *a means of describing the state of an accounting system* at any instant. Now this is most naturally accomplished by listing the "values" of the various accounts in the system.

So the first step must be to identify an algebraic structure to which the account values will belong.The structure chosen must be sufficiently rich to accommodate the operations that one would expect to apply to the accounting system.

**"Ordered integral domains are the natural candidates"**

The list of account values can be conveniently displayed as a column vector.

The list of account values can be conveniently displayed as a column vector.

Such column vectors will have the special property that *the sum of all their entries is zero*, which reflects the requirement that the accounting system should always be in balance.

The list of account values can be conveniently displayed as a column vector.

Such column vectors will have the special property that *the sum of all their entries is zero*, which reflects the requirement that the accounting system should always be in balance.

Column vectors with entry sum equal to zero are called *balance vectors*.

What formal properties should account values actually have?

What formal properties should account values actually have?

One obvious requirement is the ability to add and subtract values.

What formal properties should account values actually have?

One obvious requirement is the ability to add and subtract values.

It should be stressed that multiplication is introduced as a mathematical device and it does not carry significance for accounting.

Naturally the addition, subtraction and multiplication should satisfy reasonable rules, by which we will mean the standard rules of arithmetic. Now it is time to make all of this precise.

Naturally the addition, subtraction and multiplication should satisfy reasonable rules, by which we will mean the standard rules of arithmetic. Now it is time to make all of this precise.

Let there be given a set $R$ together with two binary operations on $R$ called addition and multiplication, denoted in the usual way, such that the following rules hold for all elements $a$, $b$, $c$ of $R$:

1. (a + b) + c = a + (b + c) (associative law);
2. a + b = b + a (commutative law);
3. R contains a zero element, written 0R or 0, such that a+0 = a for all a in R;
4. each element a of R has a negative -a of R, with the property that a + (-a) = 0;
5. (ab)c = a(bc) (associative law);
6. ab = ba (commutative law);
7. a(b + c) = ab + ac (distributive law);
8. R contains an identity element, written 1R or 1, such that a1 = a for all a in A.

The values of the accounts in an accounting system will be elements of a commutative ring with identity R.

The values of the accounts in an accounting system will be elements of a commutative ring with identity R. However the structure of $R$ is still not rich enough. For it is an essential feature of an accounting system that the value of an account can be regarded as positive or negative (or zero, of course): here the standard convention is that the value of an account representing an asset should normally be positive, while the value of a liability account should be negative: other accounts such as profit or loss could have positive or negative values. In any event we recognize that the ring R must admit the concept of “positive” and “negative” elements. This calls for the introduction of an order relation on the domain R.

A commutative ring with identity R is said to be *linearly ordered* if there is a non-empty subset P of R not containing 0, called the set of positive elements, such that the following conditions are satisfied:

9. if $a, b \in P$, then $a + b \in P$ and $ab \in P$;
10. for each $a \in R$, one of the following holds: $a \in P$, $a = 0$, $-a \in P$.

A commutative ring with identity R is said to be *linearly ordered* if there is a non-empty subset P of R not containing 0, called the set of positive elements, such that the following conditions are satisfied:

9. if $a, b \in P$, then $a + b \in P$ and $ab \in P$;
10. for each $a \in R$, one of the following holds: $a \in P$, $a = 0$, $-a \in P$.

$$a < b \Longleftrightarrow b - a \in P$$

A commutative ring with identity R is said to be *linearly ordered* if there is a non-empty subset P of R not containing 0, called the set of positive elements, such that the following conditions are satisfied:

9. if $a, b \in P$, then $a + b \in P$ and $ab \in P$;
10. for each $a \in R$, one of the following holds: $a \in P$, $a = 0$, $-a \in P$.

$$a < b \Longleftrightarrow b - a \in P$$

$$R \setminus (P \cup \{0\})$$

A commutative ring with identity R is said to be *linearly ordered* if there is a non-empty subset P of R not containing 0, called the set of positive elements, such that the following conditions are satisfied:

9. if $a, b \in P$, then $a + b \in P$ and $ab \in P$;
10. for each $a \in R$, one of the following holds: $a \in P$, $a = 0$, $-a \in P$.

$$a < b \Longleftrightarrow b - a \in P$$

$$R \setminus (P \cup \{0\})$$

11. for any $a, b \in R$, exactly one of the statements $a < b$, $a = b$, $b < a$ holds.

A commutative ring with identity R is said to be *linearly ordered* if there is a non-empty subset P of R not containing 0, called the set of positive elements, such that the following conditions are satisfied:

9. if $a, b \in P$, then $a + b \in P$ and $ab \in P$;
10. for each $a \in R$, one of the following holds: $a \in P$, $a = 0$, $-a \in P$.

$$a < b \Longleftrightarrow b - a \in P$$

$$R \setminus (P \cup \{0\})$$

11. for any $a, b \in R$, exactly one of the statements $a < b$, $a = b$, $b < a$ holds.

12. if $ab = 0$ with $a, b \in R$, then $a = 0$ or $b = 0$. A

$$\mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix}, \quad v_i \in R$$

So far arbitrary column vectors over an ordered domain R have been considered. Now our aim is to use such vectors to express the state of an accounting system by listing the balances of the various accounts as the entries of the vector.

So far arbitrary column vectors over an ordered domain R have been considered. Now our aim is to use such vectors to express the state of an accounting system by listing the balances of the various accounts as the entries of the vector.

However, the vectors to be used must have the property that the sum of their entries is zero.

The accounts of a company generally fall into three categories:

1. *Asset accounts*, which represent anything owned by the company;

The accounts of a company generally fall into three categories:

1. *Asset accounts*, which represent anything owned by the company;
2. *Liability accounts*, which record what is owed by the company to external entities;

The accounts of a company generally fall into three categories:

1. *Asset accounts*, which represent anything owned by the company;
2. *Liability accounts*, which record what is owed by the company to external entities;
3. *Equity accounts or a profit and loss account*; these show what is owed by the company to the owners and also show the net assets of the company.

The accounts of a company generally fall into three categories:

1. *Asset accounts*, which represent anything owned by the company;
2. *Liability accounts*, which record what is owed by the company to external entities;
3. *Equity accounts or a profit and loss account*; these show what is owed by the company to the owners and also show the net assets of the company.

It is a fundamental fact that a double entry accounting system must always be in balance, a fact which is implied by the accounting equation

$$A - L = E \;\; (A - L - E = 0)$$

where A, L and E are respectively the totals of all amounts in asset accounts, liability accounts and equity or profit and loss accounts.

$$\sigma : R^n \longrightarrow R$$
$$\sigma(\mathbf{v}) = \sum_{i=1}^{n} v_i$$

$$\sigma : R^n \longrightarrow R$$

$$\sigma(\mathbf{v}) = \sum_{i=1}^{n} v_i$$

R-module homomorphism

$$\sigma : R^n \longrightarrow R$$

$$\sigma(\mathbf{v}) = \sum_{i=1}^{n} v_i$$

R-module homomorphism

The reason for introducing the function $\sigma$ is that the vectors $\mathbf{v}$ which satisfy $\sigma(\mathbf{v}) = 0$ are exactly the balance vectors in $R^n$.

$$\sigma : R^n \longrightarrow R$$
$$\sigma(\mathbf{v}) = \sum_{i=1}^{n} v_i$$

R-module homomorphism

The reason for introducing the function $\sigma$ is that the vectors $\mathbf{v}$ which satisfy $\sigma(\mathbf{v}) = 0$ are exactly the balance vectors in $R^n$.

We shall write $Bal_n(R)$ for the set of all balance vectors in $R^n$, so that $Ker(\sigma) = Bal_n(R)$ is a submodule of $R^n$, which will be called the *balance module* of degree $n$ over $R$.

# Examples of balance vectors

An especially important type of balance vector occurs when there are *just two non-zero entries*, one of which will of course have to be the negative of the other. Such vectors are called *simple balance vectors*.

# Examples of balance vectors

An especially important type of balance vector occurs when there are *just two non-zero entries*, one of which will of course have to be the negative of the other. Such vectors are called *simple balance vectors*.

As an example of a simple balance vector, consider

$$\mathbf{e}(i, j), i \neq j$$

which is the vector in $R^n$, $(n \geq 2)$, whose ith entry is 1 and jth entry is -1, with all other entries 0. Then $\mathbf{e}(i, j)$ is a simple balance vector in $R^n$.

# Examples of balance vectors

An especially important type of balance vector occurs when there are *just two non-zero entries*, one of which will of course have to be the negative of the other. Such vectors are called *simple balance vectors*.

As an example of a simple balance vector, consider

$$\mathbf{e}(i,j), i \neq j$$

which is the vector in $R^n$, $(n \geq 2)$, whose ith entry is 1 and jth entry is -1, with all other entries 0. Then $\mathbf{e}(i,j)$ is a simple balance vector in $R^n$.

The $\mathbf{e}(i,j)$ are called *elementary balance vectors*.

**Theorem**

*Let $R$ be an ordered domain and let $n > 1$ be an integer. Then the elementary balance vectors $\boldsymbol{e}(1,2), \boldsymbol{e}(2,3), \ldots, \boldsymbol{e}(n-1,n)$ constitute an $R$-basis of $Bal_n(R)$. Thus $Bal_n(R)$ is a free $R$-module of rank $n-1$.*

Let $\mathbf{u}$ be a vector in $R^n$ whose entry sum equals 1. Denote by $R\mathbf{u}$ the set of all multiples of $\mathbf{u}$ by elements of R. Then $R\mathbf{u}$ is a submodule of $R^n$ and in fact it is a free R-module of rank 1.

**Theorem**

*Let $R$ be an ordered domain and let $n > 1$ be an integer. Then the elementary balance vectors $\boldsymbol{e}(1,2), \boldsymbol{e}(2,3), \ldots, \boldsymbol{e}(n-1,n)$ constitute an $R$-basis of $Bal_n(R)$. Thus $Bal_n(R)$ is a free $R$-module of rank $n-1$.*

Let $\mathbf{u}$ be a vector in $R^n$ whose entry sum equals 1. Denote by $R\mathbf{u}$ the set of all multiples of $\mathbf{u}$ by elements of R. Then $R\mathbf{u}$ is a submodule of $R^n$ and in fact it is a free R-module of rank 1.

**Theorem**

*The $R$-module $R^n$ is the direct sum of the submodules $Bal_n(R)$ and $R\mathbf{u}$, i.e., $R^n = Bal_n(R) \oplus R\mathbf{u}$, where $\mathbf{u}$ is any vector such that $\sum_{i=1}^{n} \mathbf{u}_i = 1$.*

From the algebraic point of view a natural way to generate balance vectors is to take an arbitrary vector in $R^n$ and subtract from it a vector obtained by permuting its entries. The resulting vector will always be a balance vector.

A natural measure of the complexity of a balance vector is the number of its non-zero entries. For any $\mathbf{v}$ in $Bal_n(R)$ define the level of $\mathbf{v}$ to be the number of non-zero entries of $\mathbf{v}$, with the convention that the zero vector has level 1. Thus the level of a balance vector is a positive integer and the zero vector is the only balance vector with level 1. Clearly, if $k$ is any integer satisfying $1 \leq k \leq n$, then $Bal_n(R)$ has vectors of level k. One can think of the balance vectors as being classified in a hierarchy of levels. At level 1 is the zero vector, at level 2 the non-zero simple balance vectors, and thereafter balance vectors of increasing complexity.

A natural measure of the complexity of a balance vector is the number of its non-zero entries. For any $\mathbf{v}$ in $Bal_n(R)$ define the level of $\mathbf{v}$ to be the number of non-zero entries of $\mathbf{v}$, with the convention that the zero vector has level 1. Thus the level of a balance vector is a positive integer and the zero vector is the only balance vector with level 1. Clearly, if $k$ is any integer satisfying $1 \leq k \leq n$, then $Bal_n(R)$ has vectors of level k. One can think of the balance vectors as being classified in a hierarchy of levels. At level 1 is the zero vector, at level 2 the non-zero simple balance vectors, and thereafter balance vectors of increasing complexity.

**Theorem**

*Let $R$ be an ordered domain and let $k, n$ be integers satisfying $1 < k \leq n$. Then there is an $R$-basis of $Bal_n(R)$ consisting of vectors of level $k$.*

# Transactions

**Question:** How one can represent changes in the state of an accounting system which *result from economic events*. Such changes occur when a transaction is applied to the system, which means that there is a flow of value between accounts of the system. Some account balances will increase and others decrease. Of course, there may be some accounts which are unaffected by the transaction.

After a transaction has been applied to an accounting system, the system must still be in balance, i.e., the sum of all the account balances is zero. This implies that the sum of all the changes in account balances due to the transaction must equal zero. So the conclusion is that the effect of a transaction on an accounting system can be represented adding a fixed balance vector to the balance vector that describes the original state. Then the sum of these vectors is the balance vector representing the state of the system after the transaction has been applied.

After a transaction has been applied to an accounting system, the system must still be in balance, i.e., the sum of all the account balances is zero. This implies that the sum of all the changes in account balances due to the transaction must equal zero. So the conclusion is that the effect of a transaction on an accounting system can be represented adding a fixed balance vector to the balance vector that describes the original state. Then the sum of these vectors is the balance vector representing the state of the system after the transaction has been applied.

The object of the present part is to give a formal treatment of transactions and their relation to balance vectors. Once established, this relationship permits the transfer of concepts and results for balance vectors to transactions.

## Definition

Let n be a positive integer, which will correspond to the number of accounts in an accounting system, and let R be an ordered domain, which will be the realm of account values. Choose and fix a balance vector $\mathbf{v} \in Bal_n(R)$. Then a function

$$\tau_{\mathbf{v}} : Bal_n(R) \longrightarrow Bal_n(R)$$

is defined by the rule

$$\tau_{\mathbf{v}}(\mathbf{x}) = \mathbf{x} + \mathbf{v} \qquad (\mathbf{x} \in Bal_n(R)).$$

The function $\tau_{\mathbf{v}}$ is called *the transaction corresponding to the balance vector* $\mathbf{v}$.

## Definition

Let n be a positive integer, which will correspond to the number of accounts in an accounting system, and let R be an ordered domain, which will be the realm of account values. Choose and fix a balance vector $\mathbf{v} \in Bal_n(R)$. Then a function

$$\tau_{\mathbf{v}} : Bal_n(R) \longrightarrow Bal_n(R)$$

is defined by the rule

$$\tau_{\mathbf{v}}(\mathbf{x}) = \mathbf{x} + \mathbf{v} \qquad (\mathbf{x} \in Bal_n(R)).$$

The function $\tau_{\mathbf{v}}$ is called *the transaction corresponding to the balance vector* $\mathbf{v}$. The set of all such transactions will be denoted by

$$Trans_n(R) = \{\tau_{\mathbf{v}} \mid \mathbf{v} \in Bal_n(R)\}.$$

$Trans_n(R)$ with functional composition is an abelian group.

$Trans_n(R)$ with functional composition is an abelian group.

There is also an R-module structure on $Trans_n(R)$: for one can define $r\tau_{\mathbf{v}}$, where $r \in R$ and $\mathbf{v} \in Bal_n(R)$, by the rule $r\tau_{\mathbf{v}} = \tau_{r\mathbf{v}}$. Thus $(\tau_{r\mathbf{v}})(\mathbf{x}) = \mathbf{x} + r\mathbf{v}$.

$Trans_n(R)$ with functional composition is an abelian group.

There is also an R-module structure on $Trans_n(R)$: for one can define $r\tau_{\mathbf{v}}$, where $r \in R$ and $\mathbf{v} \in Bal_n(R)$, by the rule $r\tau_{\mathbf{v}} = \tau_{r\mathbf{v}}$. Thus $(\tau_{r\mathbf{v}})(\mathbf{x}) = \mathbf{x} + r\mathbf{v}$.

**Theorem**

*The assignment $\mathbf{v} \mapsto \tau_{\mathbf{v}}$ determines a function*

$$\tau : Bal_n(R) \longrightarrow Trans_n(R)$$

*which is an isomorphism of R-modules.*

**Theorem**

*Let $n, k$ be integers satisfying $1 < k \leq n$. Then $Trans_n(R)$ has an $R$-basis consisting of transactions of level $k$.*

**Theorem**

*Let $n, k$ be integers satisfying $1 < k \leq n$. Then $Trans_n(R)$ has an $R$-basis consisting of transactions of level $k$.*

**Theorem**

*Every transaction on a set of $n$ accounts is the composite of a sequence of transactions of level $k$ where $k$ is any integer such that $1 < k \leq n$.*

**Theorem**

*Let $n, k$ be integers satisfying $1 < k \leq n$. Then $Trans_n(R)$ has an $R$-basis consisting of transactions of level $k$.*

**Theorem**

*Every transaction on a set of $n$ accounts is the composite of a sequence of transactions of level $k$ where $k$ is any integer such that $1 < k \leq n$.*

**Theorem**

*Every transaction is a composite of simple transactions.*

The history of an accounting system over some period of time can be described by listing the T-diagrams for the accounts. An alternative way to describe this history is by listing in order the successive balance vectors of the system after each transaction has been applied. These balance vectors can be used as the columns of a matrix M. Since the matrix M determines the balance sheet of the company, we call it the balance matrix of the accounting system.

The history of an accounting system over some period of time can be described by listing the T-diagrams for the accounts. An alternative way to describe this history is by listing in order the successive balance vectors of the system after each transaction has been applied. These balance vectors can be used as the columns of a matrix M. Since the matrix M determines the balance sheet of the company, we call it the balance matrix of the accounting system.

Notice that we can recover the transactions which have been applied to the system from the balance matrix M by subtracting successive columns.

let $\mathbf{b}(0)$ be the initial balance vector of the system and let the successive balance vectors after application of k transactions $\mathbf{v}(1), \mathbf{v}(2), \ldots, \mathbf{v}(k)$ be $\mathbf{b}(1), \mathbf{b}(2), \ldots, \mathbf{b}(k)$, thus

$$\mathbf{b}(i) = \mathbf{b}(i-1) + \mathbf{v}(i)$$

and the balance matrix over the period is

$$M = [\mathbf{b}(0), \mathbf{b}(1), \ldots, \mathbf{b}(k)].$$

Then we recover the transactions from the matrix M from the equations

$$\mathbf{v}(i) = \mathbf{b}(i) - \mathbf{b}(i-1).$$

Let $n$ be a positive integer and $R$ an ordered domain. The type of a balance vector $\mathbf{v} \in Bal_n(R)$,

$$type(\mathbf{v})$$

is the $n$-column vector whose ith entry is 0, $+$ or $-$ according as $v_i = 0$, $v_i > 0$ or $v_i < 0$ respectively.

Let $n$ be a positive integer and $R$ an ordered domain. The type of a balance vector $\mathbf{v} \in Bal_n(R)$,

$$type(\mathbf{v})$$

is the $n$-column vector whose ith entry is 0, + or $-$ according as $v_i = 0$, $v_i > 0$ or $v_i < 0$ respectively.

The type of a transaction $\tau_{\mathbf{v}}$ is then defined to be the type of $\mathbf{v}$.

Let $\mathbf{s}$ and $\mathbf{t}$ be types of balance vectors in $Bal_n(R)$, thus the entries of the vectors $\mathbf{s}, \mathbf{t}$ are $0, +$ or $-$. A binary relation $\leq$ on the set of types of vectors in $Bal_n(R)$ is defined as follows:

$$\mathbf{s} \leq \mathbf{t}$$

is to mean that $s_i = t_i$ or $s_i = 0$ for $i = 1, 2, \ldots, n$.

As is usually done with a partial order, one can visualize the partially ordered set of types by means of its *Hasse diagram*, in which the least complex types occur lower down in the diagram. At the lowest point will be the type of the zero vector 0, which consists entirely of zeros, while $type(\mathbf{v})$ sits directly below $type(\mathbf{w})$ if the entries of $type(\mathbf{v})$ and $type(\mathbf{w})$ are the same except that $type(\mathbf{v})$ has one more zero entry.

As is usually done with a partial order, one can visualize the partially ordered set of types by means of its *Hasse diagram*, in which the least complex types occur lower down in the diagram. At the lowest point will be the type of the zero vector 0, which consists entirely of zeros, while $\mathit{type}(\mathbf{v})$ sits directly below $\mathit{type}(\mathbf{w})$ if the entries of $\mathit{type}(\mathbf{v})$ and $\mathit{type}(\mathbf{w})$ are the same except that $\mathit{type}(\mathbf{v})$ has one more zero entry.

**Level**

The level of a transaction is defined to be the level of its associated balance vector, i.e., it is the number of $+$ and $-$ signs in the type.

As is usually done with a partial order, one can visualize the partially ordered set of types by means of its *Hasse diagram*, in which the least complex types occur lower down in the diagram. At the lowest point will be the type of the zero vector 0, which consists entirely of zeros, while $type(\mathbf{v})$ sits directly below $type(\mathbf{w})$ if the entries of $type(\mathbf{v})$ and $type(\mathbf{w})$ are the same except that $type(\mathbf{v})$ has one more zero entry.

**Level**

The level of a transaction is defined to be the level of its associated balance vector, i.e., it is the number of $+$ and $-$ signs in the type.

The concept of level permits a linear ordering of transaction types in which types of small level occur further down in the partial ordering of types. Clearly the highest possible level is n and the lowest level is 1, the type of the identity transaction. Thus the level is to be regarded as a measure of the complexity of a transaction type.

There are 13 transaction types in $Trans_3(R)$. These are listed below in descending order of levels:

$$\text{level 3}: \quad \boxed{\begin{bmatrix} + \\ + \\ - \end{bmatrix}}, \begin{bmatrix} - \\ + \\ + \end{bmatrix}, \begin{bmatrix} + \\ - \\ + \end{bmatrix}, \begin{bmatrix} - \\ - \\ + \end{bmatrix}, \begin{bmatrix} + \\ - \\ - \end{bmatrix}, \begin{bmatrix} - \\ + \\ - \end{bmatrix}$$

$$\text{level 2}: \quad \begin{bmatrix} + \\ - \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ + \\ - \end{bmatrix}, \begin{bmatrix} - \\ 0 \\ + \end{bmatrix}, \begin{bmatrix} - \\ + \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ - \\ + \end{bmatrix}, \boxed{\begin{bmatrix} + \\ 0 \\ - \end{bmatrix}}$$

$$\text{level 1}: \quad \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

**Theorem**

*Let $n$ be an integer greater than 1.*

1. *The number of $n$-transaction types of level $r$ is $\binom{n}{r}(2^r - 2)$, where $1 < r \leq n$.*
2. *The total number of $n$-transaction types is $3^n - 2^{n+1} + 2$.*

**Theorem**

*Let $n$ be an integer greater than 1.*

1. *The number of $n$-transaction types of level $r$ is $\binom{n}{r}(2^r - 2)$, where $1 < r \leq n$.*
2. *The total number of $n$-transaction types is $3^n - 2^{n+1} + 2$.*

**Theorem**

*Let $n$ and $r$ be integers such that $1 \leq r \leq n$. Then*

1. *the number of $n$-transaction types with exactly $r$ entries $+$, i.e., debits, is $\binom{n}{r}(2^{n-r} - 1)$, and this is also the number of types with exactly $r$ entries $-$, i.e., credits.*
2. *he number of $n$-transaction types with exactly $r$ zeros (i.e., $r$ unaffected accounts) is $\binom{n}{r}(2^{n-r} - 2)$ if $r < n$ and 1 if $r = n$.*

Suppose that $\mathbf{v} \in Bal_n(R)$ where $R$ is an ordered domain, and regard $\mathbf{v}$ as a transaction vector. We define a corresponding digraph D with vertex set the set of accounts $\{a_1, a_2, \ldots, a_n\}$. An edge $\langle a_i, a_j \rangle$ is to be drawn from account $a_i$ to account $a_j$ if $v_i < 0$ and $v_j > 0$, i.e., the transaction debits $a_j$ and credits $a_i$, while it may also affect other accounts.

Suppose that $\mathbf{v} \in Bal_n(R)$ where $R$ is an ordered domain, and regard $\mathbf{v}$ as a transaction vector. We define a corresponding digraph D with vertex set the set of accounts $\{a_1, a_2, \ldots, a_n\}$. An edge $\langle a_i, a_j \rangle$ is to be drawn from account $a_i$ to account $a_j$ if $v_i < 0$ and $v_j > 0$, i.e., the transaction debits $a_j$ and credits $a_i$, while it may also affect other accounts.

It follows at once from the definition that the digraph of a transaction has the special property that no vertex can have positive in-degree and positive out-degree. In fact a rather stronger property holds.

Suppose that $\mathbf{v} \in Bal_n(R)$ where $R$ is an ordered domain, and regard $\mathbf{v}$ as a transaction vector. We define a corresponding digraph D with vertex set the set of accounts $\{a_1, a_2, \ldots, a_n\}$. An edge $\langle a_i, a_j \rangle$ is to be drawn from account $a_i$ to account $a_j$ if $v_i < 0$ and $v_j > 0$, i.e., the transaction debits $a_j$ and credits $a_i$, while it may also affect other accounts.

It follows at once from the definition that the digraph of a transaction has the special property that no vertex can have positive in-degree and positive out-degree. In fact a rather stronger property holds.

**Theorem**

*If $D$ is the digraph of a transaction vector, then the vertex set of $D$ is the union of three disjoint subsets $V_0, V_1, V_2$ such that there is an edge from each vertex of $V_1$ to each vertex of $V_2$ and no other edges are present in $D$.*

**Theorem**

*There is a bijective function from the set of all types of balance vectors with $n$ entries and the set of all digraphs $D$ on a given $n$-element vertex set with the property that the vertex set of $D$ is the union of three disjoint subsets $V_0, V_1, V_2$ such that there is an edge from each vertex of $V_1$ to each vertex of $V_2$ and no other edges are present in $D$.*

**Theorem**

*There is a bijective function from the set of all types of balance vectors with $n$ entries and the set of all digraphs $D$ on a given $n$-element vertex set with the property that the vertex set of $D$ is the union of three disjoint subsets $V_0, V_1, V_2$ such that there is an edge from each vertex of $V_1$ to each vertex of $V_2$ and no other edges are present in $D$.*

**Theorem**

*The number of digraphs $D$ with a fixed set of $n$ vertices such that the vertex set is the union of three disjoint subsets $V_0, V_1, V_2$ and there is an edge from each vertex of $V_1$ to each vertex of $V_2$ while no other edges are present in $D$ is equal to $3^n - 2^{n+1} + 2$.*

# Abstract Accounting Systems

In the last two parts it has been shown how one can represent the state of an accounting system by a balance vector and a change in the state of the system by a transaction vector, which is itself a balance vector.

In the last two parts it has been shown how one can represent the state of an accounting system by a balance vector and a change in the state of the system by a transaction vector, which is itself a balance vector. Now an essential component of any accounting system is the set of rules by which the system operates. The next objective is to complete the definition of our basic model of an accounting system by specifying in algebraic form the rules that govern the operation of the system.

In any real life accounting system there will be certain transactions that would be regarded as improper. A transaction might be contrary to sound business practice or it might violate government regulations.

For example, a transaction that leads to a transfer of funds from an employee's pension account to cash would not be permitted under normal circumstances. To exclude such undesirable operations, an accounting system should come equipped with a list of transactions that are regarded as valid operations for the system. These will be called allowable transactions.

Another feature of an accounting system is that, even if a transaction is allowable, its application might still be rejected if it caused an unacceptable balance to appear in some account.

For example, in the case of a retail firm customer credit accounts are likely to have limits. A purchase on credit by a customer would not be permitted if it led to a balance that exceeded the customer's credit limit. There may also be minimum balances for certain reserve accounts in an accounting system. Thus one recognizes the existence of allowable balances, as well as allowable transactions.

Before a transaction is accepted by an accounting system, it must first be screened for allowability. Should it pass this test, the balance vector which results when the transaction is applied must be computed. If the new balance vector is allowable, the transaction is approved and applied to the system.

Before a transaction is accepted by an accounting system, it must first be screened for allowability. Should it pass this test, the balance vector which results when the transaction is applied must be computed. If the new balance vector is allowable, the transaction is approved and applied to the system.

The basic model of an accounting system should include the following:

Before a transaction is accepted by an accounting system, it must first be screened for allowability. Should it pass this test, the balance vector which results when the transaction is applied must be computed. If the new balance vector is allowable, the transaction is approved and applied to the system.

The basic model of an accounting system should include the following:

1. a set of accounts in a specified order;

Before a transaction is accepted by an accounting system, it must first be screened for allowability. Should it pass this test, the balance vector which results when the transaction is applied must be computed. If the new balance vector is allowable, the transaction is approved and applied to the system.

The basic model of an accounting system should include the following:

1. a set of accounts in a specified order;
2. a set of allowable transactions;

Before a transaction is accepted by an accounting system, it must first be screened for allowability. Should it pass this test, the balance vector which results when the transaction is applied must be computed. If the new balance vector is allowable, the transaction is approved and applied to the system.

The basic model of an accounting system should include the following:

1. a set of accounts in a specified order;
2. a set of allowable transactions;
3. a set of allowable balance vectors.

The first component of the definition is an n-element set $A$ called the set of accounts. Next we introduce the notion of a balance function from A to R: this is a function

$$\alpha : A \to R$$

such that

$$\sum_{a \in A} \alpha(a) = 0_R.$$

The first component of the definition is an n-element set $A$ called the set of accounts. Next we introduce the notion of a balance function from A to R: this is a function

$$\alpha : A \to R$$

such that

$$\sum_{a \in A} \alpha(a) = 0_R.$$

To complete the specification of the model, we choose two sets of balance functions from A to R, say $T$ and $B$, where B must not be empty.

The first component of the definition is an n-element set $A$ called the set of accounts. Next we introduce the notion of a balance function from A to R: this is a function

$$\alpha : A \to R$$

such that

$$\sum_{a \in A} \alpha(a) = 0_R.$$

To complete the specification of the model, we choose two sets of balance functions from A to R, say $T$ and $B$, where B must not be empty.

The triple $\mathcal{A} = (A|T|B)$ is called *an abstract accounting system* on A over R, with the sets T and B determining respectively the transactions which may be applied and the account balances which may arise, in a manner which will be described.

$$\{a_1, a_2, \ldots, a_n\}$$

$$\{a_1, a_2, \ldots, a_n\}$$

$$\begin{bmatrix} \alpha(a_1) \\ \alpha(a_2) \\ \vdots \\ \alpha(a_n) \end{bmatrix}$$

$$\{a_1, a_2, \ldots, a_n\}$$

$$\begin{bmatrix} \alpha(a_1) \\ \alpha(a_2) \\ \vdots \\ \alpha(a_n) \end{bmatrix}$$

T and B may be regarded as sets of n-balance vectors over R, i.e., as subsets of $Bal_n(R)$. We call T and B the sets of allowable transactions and allowable balances of $\mathcal{A}$.

### The mode of operation of the accounting system

The system has an initial balance vector $\mathbf{b}(0) \in B$. A sequence of allowable transactions $\mathbf{v}(1), \mathbf{v}(2), \ldots, \mathbf{v}(m)$ is applied to the system, producing successive allowable balance vectors $\mathbf{b}(1), \mathbf{b}(2), \ldots, \mathbf{b}(m)$ where

$$\mathbf{b}(i+1) = \mathbf{b}(i) + \mathbf{v}(i),$$

provided that $\mathbf{b}(i+1)$ is allowable, i.e., it belongs to $B$: if this is not the case, then $\mathbf{b}(i+1) = \mathbf{b}(i)$.

In practice the allowable transactions will be of two sorts. There may be specific allowable transactions with fixed entries, for example, fixed rent or mortgage payments. Then there may be entire types of transactions that are allowable: a transaction in a retail firm which debits cash and credits inventory and profit/loss would be of this type. It is therefore reasonable to replace the set T by two sets $T_0$ and $T_1$ and write

$$\mathcal{A} = (A|T_0, T_1|B)$$

where $T_0$ is the list of allowable transaction types and $T_1$ is the list of specific allowable transactions.

Consider an accounting system

$$\mathcal{A} = (A|\,T|B)$$

on n accounts $a_1, \ldots, a_n$. The digraph of $\mathcal{A}$ has vertex set

$$A = \{a_1, \ldots, a_n\}$$

and an edge

$$\langle a_j, a_i \rangle$$

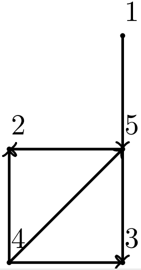
is drawn from $a_j$ to $a_i$ if some allowable transaction has its $i$th entry positive and its $j$th entry negative. Notice that the direction of the edge is from negative to positive: thus an edge $\langle a_j, a_i \rangle$ indicates a potential flow of value from account $a_j$ to account $a_i$.

# Example

$$\begin{bmatrix} - \\ 0 \\ 0 \\ 0 \\ + \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ - \\ + \end{bmatrix}, \begin{bmatrix} 0 \\ + \\ + \\ - \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 200 \\ 100 \\ 0 \\ -300 \end{bmatrix}$$

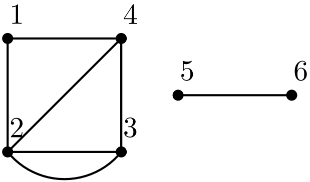
# Example

$$\begin{bmatrix} - \\ 0 \\ 0 \\ 0 \\ + \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ - \\ + \end{bmatrix}, \begin{bmatrix} 0 \\ + \\ + \\ - \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 200 \\ 100 \\ 0 \\ -300 \end{bmatrix}$$



$$\begin{bmatrix} - \\ + \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} + \\ 0 \\ + \\ - \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ + \\ - \end{bmatrix}, \begin{bmatrix} -100 \\ 200 \\ -100 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ -100 \\ 50 \\ 50 \\ 0 \\ 0 \end{bmatrix}.$$

$$\begin{bmatrix} - \\ + \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} + \\ 0 \\ + \\ - \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ + \\ - \end{bmatrix}, \begin{bmatrix} -100 \\ 200 \\ -100 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ -100 \\ 50 \\ 50 \\ 0 \\ 0 \end{bmatrix}.$$



**Theorem**

Let D be a digraph with n vertices which has no loops. Then there is an accounting system on n accounts over any ordered domain R with digraph D.

Consider two accounting systems with the same set of accounts and over the same ordered domain,

$$\mathcal{A} = (A|T|B) \text{ and } \mathcal{A}' = (A|T'|B')$$

Then $\mathcal{A}$ and $\mathcal{A}'$ are said to be *equivalent* if they have the same sets of feasible transactions.

Consider two accounting systems with the same set of accounts and over the same ordered domain,

$$\mathcal{A} = (A|T|B) \text{ and } \mathcal{A}' = (A|T'|B')$$

Then $\mathcal{A}$ and $\mathcal{A}'$ are said to be *equivalent* if they have the same sets of feasible transactions.

A transaction which is the composite of a finite sequence of allowable transactions is called a *feasible transaction* for A. Allowable transactions are feasible, but the converse need not be true.

Consider an accounting system with n accounts over an ordered domain R

$$\mathcal{A} = (A|T|B)$$

with the usual notation and conventions. An accounting system $\mathcal{A}' = (A'|T'|B')$ over R is said to be a subaccounting system of $\mathcal{A}$ if the following conditions are satisfied:

1. $A' \subseteq A$;
2. if $v \in T \cup B$, then the restriction $\mathbf{v}|_{A'}$ of $\mathbf{v}$ to $A'$ is a balance vector;
3. $T' = \{\mathbf{v}|_{A'} \mid \mathbf{v} \in T, \operatorname{sppt}(\mathbf{v}) \subseteq A'\}$;
4. $B' = \{\mathbf{b}|_{A'} \mid \mathbf{b} \in B\}$.

Consider an accounting system with n accounts over an ordered domain R

$$\mathcal{A} = (A|\,T|B)$$

with the usual notation and conventions. An accounting system $\mathcal{A}' = (A'|\,T'|B')$ over R is said to be a subaccounting system of $\mathcal{A}$ if the following conditions are satisfied:

1. $A' \subseteq A$;
2. if $v \in T \cup B$, then the restriction $\mathbf{v}|_{A'}$ of $\mathbf{v}$ to $A'$ is a balance vector;
3. $T' = \{\mathbf{v}|_{A'} \mid \mathbf{v} \in T, \mathrm{sppt}(\mathbf{v}) \subseteq A'\}$;
4. $B' = \{\mathbf{b}|_{A'} \mid \mathbf{b} \in B\}$.

$$\mathrm{sppt}(\mathbf{v}) = \{a_i \mid v_i \neq 0\}.$$

A subsystem of a system $\mathcal{A}$ with fewer accounts than $\mathcal{A}$ is called a *proper subsystem* of $\mathcal{A}$.

**Theorem**

An accounting system $\mathcal{A} = (A|T|B)$ has a proper subsystem if and only if there is a proper non-empty subset $A'$ of $A$ such that $\mathbf{v}|_{A'}$ is a balance vector whenever $v \in T \cup B$.

# Example

$$\begin{bmatrix} 50 \\ 0 \\ -50 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ -50 \\ 0 \\ 50 \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \\ -x \\ -y \end{bmatrix}, 0 \leq x, y \leq 200.$$

This has a proper subsystem on accounts $a_1, a_3$ with one allowable transaction vector $\begin{bmatrix} 50 \\ -50 \end{bmatrix}$ and allowable balance vectors $\begin{bmatrix} x \\ -x \end{bmatrix}$ where $0 \leq x \leq 200$.

# Example

$$\begin{bmatrix} 50 \\ 0 \\ -50 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ -50 \\ 0 \\ 50 \end{bmatrix}, \begin{bmatrix} -50 \\ 50 \\ 50 \\ -50 \end{bmatrix},$$

$$\begin{bmatrix} x \\ -x \\ y \\ -y \end{bmatrix}, 0 \leq x \leq 200.$$

This accounting system has no proper subsystems.

# Furthur Concepts

- Joins of accounting systems

# Furthur Concepts

- Joins of accounting systems
- Decomposable accounting systems

# Furthur Concepts

- Joins of accounting systems
- Decomposable accounting systems
- Quotients of accounting systems

# Furthur Concepts

- Joins of accounting systems
- Decomposable accounting systems
- Quotients of accounting systems
- The partial ordering of quotient systems

# Furthur Concepts

- Joins of accounting systems
- Decomposable accounting systems
- Quotients of accounting systems
- The partial ordering of quotient systems
- Lattice

# Furthur Concepts

- Joins of accounting systems
- Decomposable accounting systems
- Quotients of accounting systems
- The partial ordering of quotient systems
- Lattice
- Homomorphisms of accounting systems

# Furthur Concepts

- Joins of accounting systems
- Decomposable accounting systems
- Quotients of accounting systems
- The partial ordering of quotient systems
- Lattice
- Homomorphisms of accounting systems
- Isomorphism theorems

# Furthur Concepts

- Joins of accounting systems
- Decomposable accounting systems
- Quotients of accounting systems
- The partial ordering of quotient systems
- Lattice
- Homomorphisms of accounting systems
- Isomorphism theorems
- Accounting systems and automata

# Furthur Concepts

- Joins of accounting systems
- Decomposable accounting systems
- Quotients of accounting systems
- The partial ordering of quotient systems
- Lattice
- Homomorphisms of accounting systems
- Isomorphism theorems
- Accounting systems and automata
- Algorithms

# Furthur Concepts

- Joins of accounting systems
- Decomposable accounting systems
- Quotients of accounting systems
- The partial ordering of quotient systems
- Lattice
- Homomorphisms of accounting systems
- Isomorphism theorems
- Accounting systems and automata
- Algorithms
- The extended model

Thanks You for Your Attention

Daryasar, Mazandaran, North of Iran